to an entire network of entities that are all in charge of controlling the access to the information stored in the system. It is envisioned that in most applications the mapper will be operated by a trusted third party. This trusted third party can be a part of the organization operating the database, but also a government body or, for example, a consumer interest group.

Detailed Description Text (38):
Application of this process not only fragments the data vertically. Since each record is stored without any reference to other associated records, data is also fragmented horizontally. We refer to this database system as fully fragmented.

Detailed Description Text (76):
In this second preferred embodiment, the stored information is again fragmented vertically. The system comprises two partial-databases, each with a distinct functionality. The first partial-database 930 (the matcher) holds individuals' identifying information (the part of a medical record that refers to the real-life identity of an individual), the other partial-database 950 (the central-database) holds only medical information (the actual medical data without reference to real-life identities).. The functionality of these entities corresponds to functionality of the partial-databases 320 of FIG. 3.

Detailed Description Text (117):
By regarding the mapper B 920 as both an update terminal (300, see FIG. 3) and query terminal (330, see FIG. 3), the mechanisms disclosed in the description of the first preferred embodiment can be applied to the central-database 950 of the second preferred embodiment, resulting in vertically fragmented storage of the medical data.

Previous Doc Next Doc Go to Doc#

First Hit Fwd Refs Previous Doc Next Doc Go to Doc#

□ Generate Collection Print

L3: Entry 1 of 3 File: USPT Jun 1, 2004

DOCUMENT-IDENTIFIER: US 6745209 B2
TITLE: Synchronization of plural databases in a database replication system

Brief Summary Text (3):
"Database Replication" is specified as the application of database deltas (i.e., the results of transactions being performed against a database) from one database in a pair to the other one. Transaction I/O (e.g., inserts, updates, and deletes) applied to one database are applied to the other database, and vice-versa (for the case of bidirectional database replication). Both databases are "live" and are receiving transactions from applications and/or end users. U.S. Pat. No. 6,122,630 (Strickler et al.), which is incorporated by reference herein, discloses a bidirectional database replication scheme for controlling transaction ping-ponging.

Drawing Description Text (3):
FIG. 1 is a schematic block diagram of a prior art bidirectional database replication system; and

Detailed Description Text (6):
The following definitions are provided to promote understanding of the invention. For clarity, the definitions are phrased with respect to a scheme that replicates only two databases. However, the scope of the invention includes schemes where replication occurs between more than two databases. Replication--duplicating the contents of at least a portion of data records held in a source database to a target database. In the narrowest sense, replication involves duplicating the entire contents and format of the data records so that the two databases are totally identical, and thus interchangeable with each other. In the broadest sense, replication as defined herein involves duplicating at least the contents of a portion of the data records, and not necessarily duplicating the format of the data records. Replication thus may involve data transformation or filtering wherein the source data is altered in some manner before being applied to the target database. The concept of replication vs. transformation of data is discussed in more detail below. Replication Data--includes both "absolute" database information (e.g., set the price field to a certain value), as well as "relative" database information (e.g., add $10 or 10% to the price field). Collector (COLL)--an object or process that reads an audit trail or other transaction log file of a first database, extracts information about specified changes to the first database (e.g., insertions, updates, deletions), and passes that information to the consumer object or process defined below. In Shadowbase.TM. executing on a COMPAQ NSK (Tandem) source, the collector reads TMF or TM/MP audit trails. In a bidirectional database replication scheme, each of the two databases has an associated collector. The extractor process shown in FIG. 1 of U.S. Pat. No. 5,745,753 (Mosher, Jr.) assigned to Tandem Computers, Inc is similar in operation to the collector. Transaction Transmitter--device or object which sends transactions posted to one database to the other database for replication in the other database. In one embodiment of the present invention, the transaction transmitter is identical to the collector. In other embodiments, the transaction transmitter performs some, but not all, of the functions of the collector. In a bidirectional database replication scheme, each of the two databases has an associated transaction transmitter. Consumer (CONS)--an object or process that takes messages about database changes that are passed by the collector object or process and applies those changes to the second database. In a bidirectional database replication scheme, each of the two databases has an associated consumer. The receiver process shown in FIG. 1 of Tandem's U.S. Pat. No. 5,745,753 is similar in concept to the consumer, except that the consumer described herein can process multi-threaded (i.e., overlapping) transactions, whereas the receiver process in the Tandem patent cannot process multi-threaded transactions. Transaction Receiver--device or object which receives transactions sent by a transaction transmitter for posting to a database. In one embodiment of the present invention, the transaction receiver is identical to the consumer. In other embodiments, the transaction receiver performs some, but not all, of the functions of the consumer. In a bidirectional

database replication scheme, each of the two databases has an associated transaction receiver. Database--in the broadest sense, a database as defined herein comprises at least one table or file of data, or a portion of a table or file of data wherein the data is typically arranged in records called rows. In a narrower sense, a database is also a collection of tables or files, that is, multiple tables or files make up a database. Replication among databases thus has different meanings depending upon how the database is defined. Consider the following examples:

Detailed Description Text (9):
3. A system includes two databases, each located remotely from the other. Each database may have one or more tables or files, and the two remotely located databases replicate themselves. Replication thus maintains the two databases (including all of their respective tables or files) in the same state. The two databases are in different physical locations, and each has a respective audit trail, collector and consumer. In a typical scenario, each database resides at a different node within a network. Table--alternative name for a database. In the preferred embodiment of the present invention, replication and copying of data is performed at the file level. However, other levels of replication/copying are within the scope of the invention, such as diskcopy-type operations which are used to create the databases 126 in FIG. 1 of Tandem's U.S. Pat. No. 5,745,753. Primary Replication--effectively, unidirectional replication from a first database to a second database. Row--effectively, a single record out of a database. A row update is an individual step defined to mean a modification (e.g., insert, update, delete) to the database. Reverse Replication--effectively, unidirectional replication from the second database to the first database. Transaction--A transaction is a unit of work consisting of one or more individual steps and/or operations to be applied to one or more local and/or remote databases as a single atomic unit of work. A characteristic of transactions is the requirement that either all steps and/or operations are applied or all are rolled back in the case of a problem so that the database(s) is always left in a consistent state. Transactions are often identified by a number or name called the transaction identifier. The transaction identifier is often, though not necessarily, unique. An example of an "individual step" would be to insert a record (row) into the database. An example of an "operation" would be the procedure which increases the price column of all rows in the database by 10%.

Detailed Description Text (18):
In the examples of the present invention described below, the first and second transaction transmitters are first and second collectors, the first and second transaction receivers are first and second consumers, and the first and second databases are first and second target tables. Also, the examples below presume that strict database replication occurs without any transformation of the contents of the data or its format. However, the scope of the invention includes bidirectional replication schemes wherein at least the contents of a portion of the data or its format are transformed.

Detailed Description Text (19):
FIG. 1 is a diagram of the infrastructure for a prior art bidirectional replication system 10 illustrated and described in U.S. Pat. No. 6,122,630. In this diagram, the two databases or target tables which must be kept in the same state are located remotely from each other at different nodes in a network. However, as discussed above, the two databases may be in the same physical state and may even represent the same database replicating to itself. Thus, the communication lines shown in FIG. 1 may be merely internal data flow paths within a single computer memory, such as a bus line.

Detailed Description Text (230):
When used for a resynchronization task, the scheme actually fixes the data that is in error. This can be a unidirectional V&V/resynchronization (make the target look like the source or make the source look like the target), or a bidirectional V&V/resynchronization (make both look like the other, given some set of business rules to determine who has the more current version of data). (These business rules are typically application defined.) This scheme has various uses, including the following uses:

Detailed Description Text (232):
(2) For resynchronizing bidirectional databases after replication has been down for awhile and it is impractical to synchronize the databases by replaying the suspended feed.

Current US Original Classification (1):

707/203

Current US Cross Reference Classification (1):
707/200

Previous Doc Next Doc Go to Doc#

First Hit Fwd Refs Previous Doc Next Doc Go to Doc#

□ Generate Collection Print

L3: Entry 2 of 3 File: USPT Dec 9, 2003

DOCUMENT-IDENTIFIER: US 6662196 B2
TITLE: Collision avoidance in bidirectional database replication

Abstract Text (1):

A bidirectional database replication system includes a plurality of nodes. Each transaction at an originating node is paused prior to a commit operation. Ready to commit tokens are sent to the other nodes in the system to determine if the other nodes are prepared for the commit operation for the paused transaction. If all of the ready to commit tokens properly return to the originating node from the other nodes, thereby indicating that the other nodes are prepared for the commit operation, then the transaction is commited. For lengthy transactions, ready to sync tokens are assigned at one or more predesignated intermediate points in the transaction, and propagate throughout the system in a similar manner. The transaction continues to execute as long as all ready to sync tokens properly return to the originating node. The pause-before-commit and sync point schemes are used to avoid collisions at any of the nodes.

Brief Summary Text (3):

"Bidirectional Database Replication" is specified as the application of database deltas (i.e., the results of transactions being performed against a database) from either of two databases in a pair to the other one. Transaction I/O (e.g., inserts, updates, and deletes) applied to one database are applied to the other database and vice-versa. Both databases are "live" and are receiving transactions from applications and/or end users. U.S. Pat. No. 6,122,630 (Strickler et al.), which is incorporated by reference herein, discloses a bidirectional database replication scheme for controlling transaction ping-ponging.

Brief Summary Text (7):

Accordingly, there is an unmet need for a collision avoidance scheme in a bidirectional database replication system that is relatively simple to implement, efficiently uses communication medium, scales efficiently and easily, prevents all types of collisions, and which does not place large demands on local application programs to perform complex node coordination duties. The present invention fulfills such a need.

Drawing Description Text (3):

FIG. 1 is a schematic block diagram of a prior art bidirectional database replication system;

Drawing Description Text (4):

FIGS. 2A and 2B, taken together, is a schematic block diagram of a bidirectional database replication system having a collision avoidance scheme in accordance with the present invention; and

Detailed Description Text (2):

A bidirectional database replication system is provided that includes a plurality of nodes. Each node includes a database, a table that stores indicia of initiated transactions that are ready to be committed, but are not yet committed, and a transaction transmitter which sends selected transactions posted to the database and table entries to one or more other nodes. Each transaction being executed in the database of an originating node is paused prior to a commit operation. Then, indicia of the initiated transactions that are ready to be committed but that are not yet committed are entered into the table at the originating node. A ready to commit token is assigned to the transaction and entered into the table at the originating node. The transaction transmitter at the originating node sends the ready to commit tokens in the table of the originating node to the one or more other nodes. At each of the one or more receiving nodes, it is determined whether the database at the receiving node is prepared for a commit operation for the transaction corresponding to the received ready to commit token. If so, then the transaction transmitter in each of the other nodes sends back (selectively ping-pongs) the ready to commit token to the originating node. The commit operation of the transaction in the database of the originating node is executed only upon receipt from each of the other nodes in the system of the ready to commit token originally sent from the originating node for the transaction. In this manner, the commit operation for each transaction in the system is paused so as to allow all of the nodes in the system to prepare for the commit and thereby avoid collisions at all of the nodes in the system.

Detailed Description Text (6):
The following definitions are provided to promote understanding of the invention. For clarity, the definitions are phrased with respect to a scheme that replicates only two databases. However, the scope of the invention includes schemes where replication occurs between more than two databases. Replication--duplicating the contents of at least a portion of data records held in a source database to a target database. In the narrowest sense, replication involves duplicating the entire contents and format of the data records so that the two databases are totally identical, and thus interchangeable with each other. In the broadest sense, replication as defined herein involves duplicating at least the contents of a portion of the data records, and not necessarily duplicating the format of the data records. Replication thus may involve data transformation or filtering wherein the source data is altered in some manner before being applied to the target database. The concept of replication vs. transformation of data is discussed in more detail below.. Collector--an object or process that reads an audit trail or other transaction log file of a first database, extracts information about specified changes to the first database (e.g., insertions, updates, deletions), and passes that information to the consumer object or process defined below. In Shadowbase.TM. (a commercially available product made by ITI, Inc., Paoli, Pa.) executing on a COMPAQ NSK (Tandem) source, the collector-reads TMF or TM/MP audit trails. In a bidirectional database replication scheme, each of the two databases has an associated collector. The extractor process shown in FIG. 1 of U.S. Pat. No. 5,745,753 (Mosher, Jr.) assigned to Tandem Computers, Inc is similar in operation to the collector. Transaction Transmitter--device or object which sends transactions posted to one database to the other database for replication in the other database. In one embodiment of the present invention, the transaction transmitter is identical to the collector. In other embodiments, the transaction transmitter performs some, but not all, of the functions of the collector. In a bidirectional database replication scheme, each of the two databases has an associated transaction transmitter. Consumer--an object or process that takes messages about database changes that are passed by the collector object or process and applies those changes to the second database. In a bidirectional database replication scheme, each of the two databases has an associated consumer. The receiver process shown in FIG. 1 of Tandem's U.S. Pat. No. 5,745,753 is similar in concept to the consumer, except that the consumer described herein can process multi-threaded (i.e., overlapping) transactions, whereas the receiver process in the Tandem patent cannot process multi-threaded transactions. Transaction Receiver--device or object which receives transactions sent by a transaction transmitter for posting to a database. In one embodiment of the present invention, the transaction receiver is identical to the consumer. In other embodiments, the transaction receiver performs some, but not all, of the functions of the consumer. In a bidirectional database replication scheme, each of the two databases has an associated transaction receiver. Database--in the broadest sense, a database as defined herein comprises at least one table or file of data, or a portion of a table or file of data wherein the data is typically arranged in records called rows. In a narrower sense, a database is also a collection of tables or files, that is, multiple tables or files make up a database. Replication among databases thus has different meanings depending upon how the database is defined. Consider the following examples: 1. A system includes a single database which has two tables or files (i.e., two sub-databases) and the database replicates to itself. Replication thus maintains the two tables or files in the same state. The tables or files are in the same physical location, and each has a respective audit trail, collector and consumer. 2. A system includes a single database which has one table or file partitioned into two parts and the database replicates to itself. The first part has a plurality of records, and the second part has a plurality of records which must be kept in the same state as the first plurality of records. Replication thus maintains the two parts of the table or file in the same state. The two parts of the table or file are in the same physical location, and each has a respective audit trail, collector and consumer. 3. A system includes two databases, each located remotely from the other. Each database may have one or more tables or files, and the two remotely located databases replicate themselves. Replication thus maintains the two databases (including all of their respective tables or files) in the same state. The two databases are in different physical locations, and each has a respective audit trail, collector and consumer. In a typical scenario, each database resides at a different node within a

network. Table--alternative name for a database. In the preferred embodiment of the present invention, replication and copying of data is performed at the file level. However, other levels of replication/copying are within the scope of the invention, such as diskcopy-type operations which are used to create the databases 126 in FIG. 1 of Tandem's U.S. Pat. No. 5,745,753. Primary Replication--effectively, unidirectional replication from a first database to a second database. Row--effectively, a single record out of a database. A row update is an individual step defined to mean a modification (e.g., insert, update, delete) to the database. Reverse Replication--effectively, unidirectional replication from the second database to the first database. Transaction--A transaction is a unit of work consisting of one or more individual steps and/or operations to be applied to one or more local and/or remote databases as a single atomic unit of work. A characteristic of transactions is the requirement that either all steps and/or operations are applied or all are rolled back in the case of a problem so that the database(s) is always left in a consistent state. Transactions are often identified by a number or name called the transaction identifier. The transaction identifier is often, though not necessarily, unique. An example of an "individual step" would be to insert a record (row) into the database. An example of an "operation" would be the procedure which increases the price column of all rows in the database by 10%.

Detailed Description Text (8):
In the examples of the present invention described below, the first and second transaction transmitters are first and second collectors, the first and second transaction receivers are first and second consumers, and the first and second databases are first and second target tables. Also, the examples below presume that strict database replication occurs without any transformation of the contents of the data or its format. However, the scope of the invention includes bidirectional replication schemes wherein at least the contents of a portion of the data or its format are transformed.

Detailed Description Text (9):
FIG. 1 is a diagram of the infrastructure for a prior art bidirectional replication system 10 illustrated and described in U.S. Pat. No. 6,122,630. In this diagram, the two databases or target tables which must be kept in the same state are located remotely from each other at different nodes in a network. However, as discussed above, the two databases may be in the same physical state and may even represent the same database replicating to itself. Thus, the communication lines shown in FIGS. 2A and 2B may be merely internal data flow paths within a single computer memory, such as a bus line.

Detailed Description Text (17):
FIGS. 2A and 2B, taken together, shows one preferred embodiment of the present invention in the form of a system 44. FIG. 2 is similar to FIG. 1, except for the addition of a ready to commit table at each node, additional communication paths between the consumers and audit trails at each node, pause logic inside the local application programs, and a ready to sync table at each node (described later on in the disclosure). Specifically, node A includes ready to commit table 46 (hereafter, "RTC table A") and node B includes ready to commit table 48 (hereafter, "RTC table B"). An input of the RTC table A is connected to the output of the consumer A, and the output of the RTC table A is connected to the input of the audit trail A. The RTC table A is also in bidirectional communication with the local application program A of the local input device A. The RTC table B is connected in a similar manner to the corresponding elements of node B.

Current US Original Classification (1):
707/202

Current US Cross Reference Classification (1):
707/204

Current US Cross Reference Classification (2):
707/8

CLAIMS:

1. A method of avoiding collisions in a bidirectional database replication system including a plurality of nodes connected via communication media in a topology, each node including (i) a database, (ii) a table that stores indicia of initiated transactions that are ready to be

committed, but are not yet committed, and (iii) a transaction transmitter, each transaction being one or more transaction steps or transaction operations, the method comprising: (a) the transaction transmitter of an originating node sending selected transactions posted to the database of the originating node to one or more other nodes for replication at the one or more other nodes; b) pausing each transaction being executed in a database of an originating node prior to a commit operation for the transaction; (c) entering into the table at the originating node, indicia of the initiated transactions that are ready to be committed but that are not yet committed, and assigning a ready to commit token to the transaction; (d) the transaction transmitter at the originating node sending the ready to commit tokens in the table of the originating node to the one or more other nodes; (e) determining at each of the one or more other nodes whether the database at the one or more other nodes is prepared for a commit operation for the transactions corresponding to each received ready to commit tokens, and, if so, the transaction transmitter in each of the other nodes sends back the ready to commit tokens to the respective originating node; and (f) executing the commit operation of the transaction in the database of the originating node only upon receipt from each of the other nodes in the system of the ready to commit token originally sent from the originating node for the transaction, wherein the commit operation for each transaction in the system is paused so as to allow all of the nodes in the system to prepare for the commit and thereby avoid collisions at all of the nodes in the system.

9. An article of manufacture for avoiding collisions in a bidirectional database replication system including a plurality of nodes connected via communication media in a topology, each node including (i) a database, (ii) a table that stores indicia of initiated transactions that are ready to be committed, but are not yet committed, and (iii) a transaction transmitter, each transaction being one or more transaction steps or transaction operations, the article of manufacture comprising a computer-readable medium holding computer-executable instructions for performing a method comprising: (a) the transaction transmitter of an originating node sending selected transactions posted to the database of the originating node to one or more other nodes for replication at the one or more other nodes; (b) pausing each transaction being executed in a database of an originating node prior to a commit operation for the transaction; (c) entering into the table at the originating node, indicia of the initiated transactions that are ready to be committed but that are not yet committed, and assigning a ready to commit token to the transaction; (d) the transaction transmitter at the originating node sending the ready to commit tokens in the table of the originating node to the one or more other nodes; (e) determining at each of the one or more other nodes whether the database at the one or more other nodes is prepared for a commit operation for the transactions corresponding to each received ready to commit tokens, and, if so, the transaction transmitter in each of the other nodes sends back the ready to commit tokens to the respective originating node; and (f) executing the commit operation of the transaction in the database of the originating node only upon receipt from each of the other nodes in the system of the ready to commit token originally sent from the originating node for the transaction, wherein the commit operation for each transaction in the system is paused so as to allow all of the nodes in the system to prepare for the commit and thereby avoid collisions at all of the nodes in the system.

17. An apparatus for avoiding collisions in a bidirectional database replication system including a plurality of nodes connected via communication media in a topology, each node including (i) a database, (ii) a table that stores indicia of initiated transactions that are ready to be committed, but are not yet committed, and (iii) a transaction transmitter, each transaction being one or more transaction steps or transaction operations, the apparatus comprising: (a) a transaction transmitter of an originating node which sends selected transactions posted to the database of the originating node to one or more other nodes for replication at the one or more other nodes; (b) means for pausing each transaction being executed in a database of an originating node prior to a commit operation for the transaction; (c) means for entering into the table at the originating node, indicia of the initiated transactions that are ready to be committed but that are not yet committed, and assigning a ready to commit token to the transaction, the transaction transmitter at the originating node sending the ready to commit tokens in the table of the originating node to the one or more other nodes; (d) means for determining at each of the one or more other nodes whether the database at the one or more other nodes is prepared for a commit operation for the transactions corresponding to each received ready to commit tokens, and, if so, the transaction transmitter in each of the other nodes sends back the ready to commit tokens to the respective originating node; and (e) means for executing the commit operation of the transaction in the database of the originating node only upon receipt from each of the other nodes in the system of the ready

to commit token originally sent from the originating node for the transaction, wherein the commit operation for each transaction in the system is paused so as to allow all of the nodes in the system to prepare for the commit and thereby avoid collisions at all of the nodes in the system.

25. A method of avoiding collisions in a bidirectional database replication system including a plurality of nodes connected via communication media in a topology, each node including (i) a database, (ii) a table that stores indicia of initiated transactions that are at one or more predesignated intermediate points in the transaction but are not yet committed, and (iii) a transaction transmitter, each transaction being one or more transaction steps or transaction operations, the method comprising: (a) the transaction transmitter of an originating node sending selections transactions posted to the database of the originating node to one or more other nodes for replication at the one or more other nodes; (b) entering into the table at the originating node, indicia of the initiated transactions that are at one or more predesignated intermediate points in the transaction but are not yet committed, and assigning a ready to sync token to the transaction at each of the predesignated intermediate points; (c) the transaction transmitter at the originating node sending the ready to sync tokens in the table of the originating node to the one or more other nodes; (d) determining at each of the one or more other nodes whether the database at the one or more other nodes is prepared to properly process the transaction up to the intermediate point associated with the respective ready to sync token, and, if so, the transaction transmitter in each of the other nodes sends back the ready to sync tokens to the respective originating node; and (e) stopping the execution of a transaction in the system if the originating node fails to receive back a ready to sync token from at least one of the other nodes in the system for any of the predesignated intermediate points, wherein the transaction continues to execute as long as all ready to sync tokens properly return to the originating node, thereby indicating that no collisions should occur at any of the nodes in the system up to the most recent intermediate point in the transaction.

29. An article of manufacture for avoiding collisions in a bidirectional database replication system including a plurality of nodes connected via communication media in a topology, each node including (i) a database, (ii) a table that stores indicia of initiated transactions that are at one or more predesignated intermediate points in the transaction but are not yet committed, and (iii) a transaction transmitter, each transaction being one or more transaction steps or transaction operations, the article of manufacture comprising a computer-readable medium holding computer-executable instructions for performing a method comprising: (a) the transaction transmitter of an originating node sending selections transactions posted to the database of the originating node to one or more other nodes for replication at the one or more other nodes; (b) entering into the table at the originating node, indicia of the initiated transactions that are at one or more predesignated intermediate points in the transaction but are not yet committed, and assigning a ready to sync token to the transaction at each of the predesignated intermediate points; (c) the transaction transmitter at the originating node sending the ready to sync tokens in the table of the originating node to the one or more other nodes; (d) determining at each of the one or more other nodes whether the database at the one or more other nodes is prepared to properly process the transaction up to the intermediate point associated with the respective ready to sync token, and, if so, the transaction transmitter in each of the other nodes sends back the ready to sync tokens to the respective originating node; and (e) stopping the execution of a transaction in the system if the originating node fails to receive back a ready to sync token from at least one of the other nodes in the system for any of the predesignated intermediate points, wherein the transaction continues to execute as long as all ready to sync tokens properly return to the originating node, thereby indicating that no collisions should occur at any of the nodes in the system up to the most recent intermediate point in the transaction.

33. An apparatus for avoiding collisions in a bidirectional database replication system including a plurality of nodes connected via communication media in a topology, each node including (i) a database, (ii) a table that stores indicia of initiated transactions that are at one or more predesignated intermediate points in the transaction but are not yet committed, and (iii) a transaction transmitter, each transaction being one or more transaction steps or transaction operations, the apparatus comprising: (a) a transaction transmitter of an originating node which sends selected transactions posted to the database of the originating node to one or more other nodes for replication at the one or more other nodes; (b) means for entering into the table at the originating node, indicia of the initiated transactions that are at one or more predesignated intermediate points in the transaction but are not yet committed,

and assigning a ready to sync token to the transaction at each of the predesignated intermediate points, the transaction transmitter at the originating node sending the ready to sync tokens in the table of the originating node to the one or more other nodes; (c) means for determining at each of the one or more other nodes whether the database at the one or more other nodes is prepared to properly process the transaction up to the intermediate point associated with the respective ready to sync token, and, if so, the transaction transmitter in each of the other nodes sends back the ready to sync tokens to the respective originating node; and (d) means for stopping the execution of a transaction in the system if the originating node fails to receive back a ready to sync token from at least one of the other nodes in the system for any of the predesignated intermediate points, wherein the transaction continues to execute as long as all ready to sync tokens properly return to the originating node, thereby indicating that no collisions should occur at any of the nodes in the system up to the most recent intermediate point in the transaction.

Previous Doc Next Doc Go to Doc#

First Hit Fwd Refs Previous Doc Next Doc Go to Doc#
End of Result Set

☑ Generate Collection Print

L3: Entry 3 of 3 File: USPT Sep 19, 2000

DOCUMENT-IDENTIFIER: US 6122630 A
TITLE: Bidirectional database replication scheme for controlling ping-ponging

Abstract Text (1):
Transaction ping-pong is selectively prevented in a bidirectional database replication system. The system has a plurality of nodes connected via communication media in a topology. Each node includes a database and a transaction transmitter or collector which sends transactions posted to the database to a database at one or more other nodes for replication in the databases of the one or more other nodes. All transactions to be posted to databases in remote nodes that were sent by a local node are detected, and the database at the local node is inhibited from posting selective transactions which were detected as being originally sent by the local node.

Brief Summary Text (2):
The present invention relates to the field of data replication. "Bidirectional Database Replication" is specified as the application of database deltas (i.e., the results of transactions being performed against a database) from either of two databases in a pair to the other one. Transaction I/O (e.g., inserts, updates, and deletes) applied to one database are applied to the other database and vice versa. Both databases are "live" and are receiving transactions from applications and/or end users. "Bidirectional Database Replication" presents itself in two ways--"bidirectional homogeneous database replication" and "bidirectional heterogeneous replication." In homogeneous database replication, the two databases in the pair are identical. In heterogeneous replication, while there must necessarily be some commonality between the two databases in the data they contain, there are differences in the databases. For example, the databases may be of the same commercial brand but have differing physical and logical structures, or the databases may be essentially identical but be of different commercial brands.

Brief Summary Text (3):
The typical purpose of "Bidirectional Homogeneous Database Replication" is to provide a higher degree of fault tolerance than can be provided through conventional unidirectional database replication from a production database to a backup database ("hot backup replication"). In traditional hot backup replication, database deltas are captured from the production database and are applied to a backup database that is usually located remotely. In the event of a failure of the primary system, the applications on the backup system must be started and the end users must be routed to the backup database. In the best implementations, this process takes several minutes. Furthermore, assuming the failure of the primary system was not caused by a severe catastrophe, the computer operators must be moved from the production location to the backup location. For example, in severe natural disasters or terrorist incidents, it may not be possible to transport the computer operators to the backup site. In typical "Bidirectional Homogeneous Database Replication," both databases (and both sites if they are remote from one another) are live at all times. All applications are running on both databases, end users are already enabled to perform transactions on either database, and each data center is fully staffed. In the event of a failure of one of the two databases, all processing is performed on the remaining live database. No downtime of the application is endured.

Brief Summary Text (4):
The typical purpose of "Bidirectional Heterogeneous Database Replication" is to allow heterogeneous computer systems and databases to share data in one enterprise with minimal impact on the production processing occurring on either system. Traditional Open Database Connectivity solutions for this problem allow applications or end users on one system to run queries against the databases on the other systems. This causes massive read activity on the other system that hampers production transaction processing. "Bidirectional Heterogeneous

Database Replication," however, uses very little system resources on either system. Changes to data on one system which are needed on the other system are replicated as they occur allowing the applications and end users on each system access to the local data in a near real-time manner. Many financial institutions perform much of their core transaction processing and all of their settlement activity on legacy mainframe systems. Most of the new applications that these institutions would like to use, however, are written for newer computer platforms and modem SQL databases. By using "Bidirectional Heterogeneous Database Replication," these institutions can utilize both their legacy systems and modem systems, databases, and applications and propagate data changes from either system/database to the other.

Brief Summary Text (5):
While the business applications of "Bidirectional Homogeneous Database Replication" and "Bidirectional Heterogeneous Database Replication" are quite different, the technology that enables each is the same. The technical difference between unidirectional database replication and "Bidirectional Database Replication" is that "Bidirectional Database Replication" can recognize the origin of a given transaction and apply the transaction from one database to the other without then necessarily applying it back to the database from which it originated.

Brief Summary Text (9):
Bidirectional replication simplifies the manual procedures necessary to manage outages of a system during planned and unplanned switchovers. These procedures are currently required due to a replication side effect, which is referred to herein as "ping-pong." That is, if you attempted to configure a "bidirectional" and live stand-by environment with two unidirectional schemes, transaction audit events would oscillate indefinitely between the two systems. This is because events applied on one system would be captured and would continually be bounced back and forth, thereby resulting in a "ping-pong" effect. Conventional unidirectional replication requires manual procedures to turn on the flow or start a stand-by copy following a fail-over (once the original primary system has come back on line as the backup system), due to operational complexities in managing this environment. Resolving the bidirectional "ping-pong" issue would then provide the capability for a "Sizzling Hot Stand-By" environment, particularly if the latency time is low. ("Latency" is defined as the time that the commit takes place on one system to be applied on the peer or other system.) Once a solution is provided to remove the "ping-pong," two-way flow is then possible. Note that it is not necessary to provide a means to detect or eliminate collisions, in the event that replication is enabled in both directions simultaneously. Collision avoidance is primarily an application-related issue that requires some method of version identification in the database record and is complimentary to bidirectional replication.

Brief Summary Text (11):
Conventional peer-to-peer data replication methodologies are bidirectional and also often have the freedom that the updates to particular data rows can occur on either system and thus involve a conflict resolution mechanism to prevent ping-pong.

Brief Summary Text (12):
Conventional peer-to-peer, bidirectional data replication systems rely on one of the following schemes:

Brief Summary Text (21):
Database partitioning is not a very practical solution, unless a user is willing to repartition their database to accommodate this approach. In general, most users are often unable to repartition their database. In fact, in some instances, there might be technical performance reasons for partitioning their database using different keys (e.g., load balancing, throughput, etc.) than the keys useful for bidirectional partitioning.

Brief Summary Text (33):
6. Pass the Book--Simple unidirectional replication can be used in a bidirectional capability if it is only turned on in one direction at a time, e.g., on every odd hour from system A to B, and on every even hour from system B to A. In this scenario, the source system must always be the system on which updates are made. This approach is not low-latency and has numerous operational problems, including the inability to fully utilize both systems at all times.

Brief Summary Text (34):
Conventional peer-to-peer schemes, including those discussed above, have a high latency, are

limited to the row update level, and/or have significant operational restrictions. Critical to all peer-to-peer, bidirectional data replication methodologies is the requirement to control updates on one system which are replicated to the other system from replicating back to the original system. This problem, presently called "ping-pong," results in an oscillation which would waste considerable computer resources and communication bandwidth and may result in data loss or corruption (a "dirty" database). Likewise, a useful replication scheme must not impact the primary system and must be restartable without data loss or corruption in the event that one or more of the replication peers are not available for a period of time. Accordingly, there is still an unmet need for a data replication scheme which avoids ping-pong, has a low latency, is restartable, and provides operational ease of use. In addition, in some instances, a limited amount of ping-pong may be useful. For example, one may need to see the "pong" reply to verify that all important transactions get applied. Accordingly, there is a need for a bidirectional data replication scheme which allows for selective ping-pong of transactions. The present invention fulfills all of the needs discussed above.

Brief Summary Text (36):
The present invention provides various schemes for selectively preventing transaction ping-pong in a bidirectional database replication system. The system includes a first database, a second database, a first transaction transmitter which sends transactions posted to the first database to the second database for replication in the second database, and a second transaction transmitter which sends transactions posted to the second database to the first database for replication in the first database. In each of the schemes, all transactions to be posted to the second database that were sent by the first transaction transmitter are detected, and the first database is inhibited from posting selective transactions which were detected as being originally sent by the first transaction transmitter.

Brief Summary Text (37):
The present invention also provides various schemes for selectively preventing transaction ping-pong in a bidirectional database replication system. The system includes a plurality of nodes connected via communication media in a topology. Each node includes a database and a transaction transmitter which sends transactions posted to the database to a database at one or more other nodes for replication in the databases of the one or more other nodes. In each of the schemes, all transactions to be posted to databases in remote nodes that were sent by a local node are detected, and the database at the local node is inhibited from posting selective transactions which were detected as being originally sent by the local node.

Drawing Description Text (4):
FIG. 2 is a schematic block diagram of a bidirectional data replication system for use with the present invention;

Drawing Description Text (5):
FIG. 3 is a schematic block diagram of a bidirectional data replication system in accordance with a first scheme of the present invention;

Drawing Description Text (7):
FIG. 5 is schematic block diagram of a bidirectional data replication system in accordance with a second scheme of the present invention;

Drawing Description Text (9):
FIG. 7 is schematic block diagram of a bidirectional data replication system in accordance with a third and a fourth scheme of the present invention;

Drawing Description Text (12):
FIG. 9 is schematic block diagram of a bidirectional data replication system in accordance with a fifth scheme of the present invention;

Drawing Description Text (14):
FIG. 11 is schematic block diagram of a bidirectional data replication system in accordance with a sixth scheme of the present invention;

Drawing Description Text (15):
FIG. 12 is schematic block diagram of a bidirectional data replication system in accordance with a seventh scheme of the present invention;

Drawing Description Text (19):
FIG. 16 is a schematic block diagram of a bidirectional data replication system in accordance with an eighth scheme of the present invention which allows for selective ping-ponging; and

Detailed Description Text (6):
Collector--an object or process that reads an audit trail or other transaction log file of a first database, extracts information about specified changes to the first database (e.g., insertions, updates, deletions), and passes that information to the consumer object or process defined below. In Shadowbase on a Tandem source, the collector reads TMF or TM/MP audit trails. In a bidirectional database replication scheme, each of the two databases has an associated collector. The extractor process shown in FIG. 1 of U.S. Pat. No. 5,745,753 (Mosher, Jr.) assigned to Tandem Computers, Inc is similar in operation to the collector.

Detailed Description Text (7):
Transaction transmitter--device or object which sends transactions posted to one database to the other database for replication in the other database. In one embodiment of the present invention, the transaction transmitter is identical to the collector. In other embodiments, the transaction transmitter performs some, but not all, of the functions of the collector. In a bidirectional database replication scheme, each of the two databases has an associated transaction transmitter.

Detailed Description Text (8):
Consumer--an object or process that takes messages about database changes that are passed by the collector object or process and applies those changes to the second database. In a bidirectional database replication scheme, each of the two databases has an associated consumer. The receiver process shown in FIG. 1 of Tandem's U.S. Pat. No. 5,745,753 is similar in concept to the consumer, except that the consumer described herein can process multi-threaded (i.e., overlapping) transactions, whereas the receiver process in the Tandem patent cannot process multi-threaded transactions.

Detailed Description Text (9):
Transaction receiver--device or object which receives transactions sent by a transaction transmitter for posting to a database. In one embodiment of the present invention, the transaction receiver is identical to the consumer. In other embodiments, the transaction receiver performs some, but not all, of the functions of the consumer. In a bidirectional database replication scheme, each of the two databases has an associated transaction receiver.

Detailed Description Text (15):
Primary replication--effectively, unidirectional replication from a first database to a second database.

Detailed Description Text (18):
Standby/Reverse replication--effectively, unidirectional replication from the second database to the first database.

Detailed Description Text (20):
Filtering--The operation of selectively choosing rows or transactions to replicate. Filtering based upon the transaction identifier is important because it allows a great degree of selectivity in comparison to the all or nothing row-based filtering schemes of prior art. Also, filtering in the present invention does not require modifications to the database or transaction log files as does the scheme used in Sybase's Replication Server Product. The net result is significantly less database space usage and an ability to have bidirectional replication to heterogeneous database types.

Detailed Description Text (23):
In the examples of the present invention described below, the first and second transaction transmitters are first and second collectors, the first and second transaction receivers are first and second consumers, and the first and second databases are first and second target tables. Also, the examples below presume that strict database replication occurs without any transformation of the contents of the data or its format. However, the scope of the invention include bidirectional replication schemes wherein at least the contents of a portion of the data or its format are transformed.

Detailed Description Text (24):
FIG. 2 is a diagram of the infrastructure for a bidirectional replication system 28 in accordance with the present invention. In this diagram, the two databases or target tables which must be kept in the same state are located remotely from each other at different nodes in a network. However, as discussed above, the two databases may be in the same physical state and may even represent the same database replicating to itself. Thus, the communication lines shown in FIG. 2 may be merely internal data flow paths within a single computer memory, such as a bus line.

Detailed Description Text (61):
FIG. 3 is a diagram of a bidirectional replication system 28' in accordance with a first embodiment of the present invention. FIG. 3 is similar to FIG. 2, except for the following differences:

Detailed Description Text (78):
FIG. 5 is a diagram of a bidirectional replication system 28" in accordance with a second, and preferred embodiment of the present invention. FIG. 5 is similar to FIG. 3, except for the following differences:

Detailed Description Text (101):
FIG. 7 is a diagram of a bidirectional replication system 28'" in accordance with a third embodiment of the present invention. FIG. 7 is similar to FIG. 2, except for the following differences:

Detailed Description Text (120):
FIG. 9 is a diagram of a bidirectional replication system 28"" in accordance with a fifth embodiment of the present invention. FIG. 9 is similar to FIG. 2, except for the following differences:

Detailed Description Text (132):
FIG. 11 is a diagram of a bidirectional replication system 28'"" in accordance with a sixth embodiment of the present invention. FIG. 11 is similar to FIG. 2, except for the following differences:

Detailed Description Text (145):
FIG. 12 is a diagram of a bidirectional replication system 28""" in accordance with a seventh embodiment of the present invention. FIG. 12 is similar to FIG. 2, except that collector A includes one or more user exits 100 and collector B includes one or more user exits 102, referred to interchangeably as "user exit A" and "user exit B," respectively. As discussed above, a user exit is customer application code that is bound in a consumer to customize data before it is applied to the target database. In this embodiment of the present invention, the user exit analyzes the contents of records of transaction data and uses the results of the analysis to decide whether to forward the records comprising the transactions to the next step in the process (here, the consumer). In this manner, the user exit may eliminate one or more of the data records or complete transactions from being passed to the consumer based upon data record content analysis.

Detailed Description Text (158):
The transaction tracking file (TRANLOG) used in scheme 2 enables a "Sizzling Hot Stand-By" solution for disaster recovery in a flexible, operating system independent method. However, due to the added overhead of this file, the interprocess message approach of scheme 6 might be implemented in some scenarios. The combination of the two options can be integrated into a solution that can tolerate a stand-by collector going down (e.g., normal shutdown or due to processor failure). For example, while the stand-by collector is operational, interprocess messages would be used from the primary consumer to identify transactions that the collector should ignore. When a stand-by collector is down, the primary consumer writes to the transaction tracking file. Once the stand-by collector is brought back up, it will identify transactions to ignore via the information in the audit trail. However, if a stand-by collector becomes non-operational after receiving a transaction identifier, the information is lost from memory. This problem may be solved by having the consumer send two messages, one at the start of the transaction and another message at the end of a transaction. If one of these messages doesn't make it to the stand-by collector, the primary system consumer could write the

information to the transaction tracking file. When the collector is restarted, it would read into memory all transaction tracking file records to identify what to ignore before reading any audit trail data. Such a solution should have the least impact on bidirectional replication performance while under the interprocess message mode of operation. It is believed that these interprocess messages, between the primary consumer and stand-by collector, should only take minimum operating resources per message.

Detailed Description Text (198):
Schemes 1-7 described above allow for bidirectional replication without ping-ponging.

Detailed Description Text (201):
FIG. 16 shows an eighth scheme of the present invention which allows for bidirectional replication with selective ping-ponging.

Current US Original Classification (1):
707/8

Current US Cross Reference Classification (1):
707/10

CLAIMS:

1. A method of selectively preventing transaction ping-pong in a bidirectional database replication system including (i) a first database and (ii) a second database, the system further including (iii) a first transaction transmitter which sends transactions posted to the first database to the second database for replication in the second database, and (iv) a second transaction transmitter which sends transactions posted to the second database to the first database for replication in the first database only after the transactions are actually posted to the second database, each transaction being one or more transaction steps or transaction operations, the method comprising:

(a) detecting all transactions to be posted to the second database that were sent by the first transaction transmitter; and

(b) inhibiting the first database from posting selective transactions which were detected as being originally sent by the first transaction transmitter, wherein any transaction sent by the second transaction transmitter is replicated in the first database only after the transaction is actually posted to the second database.

30. A method of selectively preventing transaction ping-pong in a bidirectional database replication system including a plurality of nodes connected via communication media in a topology, each node including a database and a transaction transmitter which sends transactions posted to the database to a database at one or more other nodes for replication in the databases of the one or more other nodes only after the transactions are actually posted to the database of the sending node, each transaction being one or more transaction steps or transaction operations, the method comprising:

(a) detecting all transactions to be posted to databases in remote nodes that were sent by a local node; and

(b) inhibiting the database at the local node from posting selective transactions which were detected as being originally sent by the local node, wherein any transaction sent by a transaction transmitter of a node is replicated in the databases of the one or more other nodes only after the transaction is actually posted to the database of the sending node.

34. A system for selectively preventing transaction ping-pong in a bidirectional database replication system which includes a plurality of nodes connected via communication media in a topology, each node including a database and a transaction transmitter which sends transactions posted to the database to a database at one or more other nodes for replication in the databases of the one or more other nodes only after the transactions are actually posted to the database of the sending node, each transaction being one or more transaction steps or transaction operations, the system comprising:

(a) means for detecting all transactions to be posted to databases in remote nodes that were sent by a local node; and

(b) means for inhibiting the database at the local node from posting selective transactions which were detected as being originally sent by the local node, wherein any transaction sent by a transaction transmitter of a node is replicated in the databases of the one or more other nodes only after the transaction is actually posted to the database of the sending node.

36. A system for selectively preventing transaction ping-pong in a bidirectional database replication system which includes (i) a first database and (ii) a second database, the system further including (iii) a first transaction transmitter which sends transactions posted to the first database to the second database for replication in the second database, and (iv) a second transaction transmitter which sends transactions posted to the second database to the first database for replication in the first database only after the transactions are actually posted to the second database, each transaction being one or more transaction steps or transaction operations, the system for selectively preventing transaction ping-pong comprising:

(a) means for detecting all transactions to be posted to the second database that were sent by the first transaction transmitter; and

(b) means for inhibiting the first database from posting selective transactions which were detected as being originally sent by the first transaction transmitter, wherein any transaction sent by the second transaction transmitter is replicated in the first database only after the transaction is actually posted to the second database.

39. A system according to claim 36 wherein the bidirectional database replication system further includes (v) a first transaction receiver which receives transactions sent by the second transaction transmitter for posting to the first database, and (vi) a second transaction receiver which receives transactions sent by the first transaction transmitter for posting to the second database, the means for inhibiting being located in the first transaction receiver.

46. A system according to claim 36 wherein the bidirectional database replication system further includes (v) a second transaction receiver which receives transactions sent by the first transaction transmitter for posting to the second database, (vi) a transaction log associated with the second database, and (vii) an audit trail of all transactions posted to the second database, the system for selectively preventing transaction ping-pong further comprising:

(c) means for assigning a transaction identifier to every transaction received by the second transaction receiver which is posted to the second database;

(d) means for posting the transactions received by second transaction receiver in the second database and creating an audit trail of the posted transactions; and

(e) means for storing the assigned transaction identifiers in the transaction log and associating the assigned transaction identifiers with the transactions in the audit trail,

wherein the second transaction transmitter sends the transactions in the audit trail for posting to the first database,

and wherein the means for detecting detects selective transactions in the audit trail which have a transaction identifier similar to a transaction identifier in the transaction log, and the means for inhibiting inhibits the first database from posting to the first database selective transactions in the audit trail which were detected as having a transaction identifier similar to a transaction identifier in the transaction log.

48. A system according to claim 47 wherein the bidirectional database replication system further includes (viii) a first transaction receiver which receives transactions sent by the second transaction transmitter for posting to the first database, (ix) a transaction log associated with the first database, and (x) an audit trail of all transactions posted to the first database, the first transaction transmitter sending the transactions in the audit trail of transactions posted to the first database for posting to the second database, the system for selectively preventing transaction ping-pong further comprising:

(f) means for assigning a transaction identifier to every transaction received by the first transaction receiver which is posted to the first database;

(g) means for posting the transactions received by first transaction receiver in the first database and creating an audit trail of the transactions posted to the first database;

(h) means for storing the assigned transaction identifiers in the transaction log associated with the first database and associating the transaction identifiers assigned by the means for assigning with the transactions in the audit trail of transactions posted to the first database;

(i) means for detecting all transactions to be posted to the first database that were sent by the second transaction transmitter; and

(j) means for inhibiting the second database from posting selective transactions which were detected as being originally sent by the second

transaction transmitter,

wherein the means for detecting detects selective transactions in the audit trail of transactions posted to the first database which have a transaction identifier similar to a transaction identifier in the transaction log associated with the first database, and the means for inhibiting inhibits the second database from posting to the second database selective transactions in the audit trail of transactions posted to the first database which were detected as having a transaction identifier similar to a transaction identifier in the transaction log associated with the first database.

49. A system according to claim 36 wherein the bidirectional database replication system further includes (v) a second transaction receiver which receives transactions sent by the first transaction transmitter for posting to the second database, (vi) a transaction log associated with the second database, and (vii) an audit trail of all transactions posted to the second database, the second transaction transmitter sending the transactions in the audit trail for posting to the first database, the system for selectively preventing transaction ping-pong further comprising:

(c) means for preassigning a block of unique transaction identifiers for transactions subsequently received by the second transaction receiver;

(d) means for storing the preassigned block of unique transaction identifiers in the audit trail and in the transaction log;

(e) means for assigning one of the preassigned unique transaction identifiers in the transaction log to each transaction subsequently received by the second transaction receiver; and

(f) means for posting the transactions received by the second transaction receiver in the second database and creating an audit trail of the posted transactions, the audit trail including the assigned transaction identifiers;

wherein the means for detecting detects selective transactions in the audit trail which have a transaction identifier similar to a preassigned transaction identifier stored in the audit trail, and the means for inhibiting inhibits the first database from posting to the first database selective transactions in the audit trail which were detected as having a transaction identifier similar to a preassigned transaction identifier stored in the audit trail.

52. A system according to claim 36 wherein the bidirectional database replication system further includes (v) a second transaction receiver which receives transactions sent by the first transaction transmitter, (vi) a second processor which posts the transactions received by the second transaction receiver to the second database, the second processor including a second processor identifier, and (vii) an audit trail of all transactions posted to the second database, the system for selectively preventing transaction ping-pong further comprising:

(c) means for posting the transactions received by the second transaction receiver in the second database and creating an audit trail of the posted transactions, the audit trail including the second processor identifier, the means for posting being located in the second processor;

wherein the second transaction transmitter sends the transactions in the audit trail to the first database for posting to the first database,

and wherein the means for detecting detects selective transactions which are in the audit trail of the second database and which include the second processor identifier, and the means for inhibiting comprises preventing the first database from posting selective transactions which were detected as being in the audit trail of the second database and which include the second processor identifier.

55. A system according to claim 36 wherein the bidirectional database replication system further includes (v) a second transaction receiver which receives transactions sent by the first transaction transmitter, the second transaction receiver including a second transaction receiver identifier, and (vi) an audit trail of all transactions posted to the second database, the system for selectively preventing transaction ping-pong further comprising:

(c) means for posting the transactions received by the second transaction receiver in the second database and creating an audit trail of the posted transactions, the audit trail including the second transaction receiver identifier,

wherein the second transaction transmitter sends the transactions in the audit trail to the first database for posting to the first database,

and wherein the means for detecting detects selective transactions which are in the audit trail of the second database and which include the second transaction receiver identifier, and the means for inhibiting prevents the first database from posting selective transactions which were detected as being in the audit trail of the second database and which include the second transaction receiver identifier.

58. A system according to claim 36 wherein the bidirectional database replication system further includes (v) an audit trail of all transactions posted to the second database, the audit trail including a first audit trail containing all transactions posted to the second database which were sent by the first transaction transmitter, and a second audit trail containing all transactions posted to the second database which were not sent by the first transaction transmitter, the system for selectively preventing transaction ping-pong further comprising:

(c) means for posting to the second database (i) all transactions sent by the first transaction transmitter and creating an audit trail of the posted transactions in the first audit trail, and (ii) all transactions not sent by the first transaction transmitter and creating an audit trail of the posted transactions in the second audit trail,

wherein the second transaction transmitter sends the transactions in the audit trail to the first database for posting to the first database,

and wherein the means for detecting comprises identifying selective transactions which are in the first audit trail of the second database, and the means for inhibiting prevents the first database from posting selective transactions which were identified as being in the first audit trail of the second database.

59. A system according to claim 58 wherein the bidirectional database replication system further includes (vi) a second transaction receiver which receives transactions sent by the first transaction transmitter and posts the transactions which it receives in the second database and creates an audit trail of the posted transactions in the first audit trail,

and wherein the second transaction transmitter sends the transactions in the audit trail to the first database for posting to the first database,

and wherein the means for inhibiting prevents the second transaction transmitter from sending to the first database selective transactions which are in the first audit trail of the second database.

62. A system according to claim 36 wherein the bidirectional database replication system further includes (v) a second transaction receiver which receives transactions sent by the first transaction transmitter for posting to the second database, and (vi) an audit trail of all transactions posted to the second database, the system for selectively preventing transaction ping-pong further comprising:

(c) means for posting the transactions received by the second transaction receiver in the second database and creating an audit trail of the posted transactions; and

(d) means for communicating the transactions received by the second transaction receiver to the second transaction transmitter,

wherein the second transaction transmitter sends the transactions in the audit trail to the first database for posting to the first database,

and wherein the means for detecting comprises detecting selective transactions in the audit trail which were communicated to the second transaction transmitter, and the means for inhibiting comprises preventing the second transaction transmitter from sending to the first database selective transactions in the audit trail which were detected as being communicated to the second transaction transmitter.

63. A system according to claim 36 wherein each of the transactions include transaction data which is usable to identify the origin of the transaction, the bidirectional database replication system further including (v) an audit trail of all transactions posted to the second database, and (vi) application code which interprets the transaction data within posted transactions for identifying the origin of the transaction, thereby identifying the originally posted database of the transaction, the system for selectively preventing transaction ping-pong firer comprising

(c) means for posting to the second database selective transactions sent by the first transaction transmitter and creating an audit trail of the posted transactions, wherein the second transaction transmitter sends the transactions in the audit trail to the first database for posting to the first database; and

(e) means for executing the application code on all transactions read out of the audit trail for sending to the first database,

wherein the means for detecting uses the application code to identify selective transactions which originated at the first database, and the means for inhibiting prevents the first database from posting selective transactions which were identified by the application code as having originated at the first database.

65. A system according to claim 35 wherein each of the transactions include transaction data which is usable to identify the origin of the transaction, the bidirectional database replication system further including application code which interprets the transaction data within posted transactions for identifying the origin of the transaction, thereby identifying the originally posted database of the transaction, and the means for inhibiting inhibits the database at the local node from posting selective transactions which were detected as being originally sent by a node having a predetermined origin.

67. An article of manufacture comprising a computer usable medium having computer readable code means therein for selectively preventing transaction ping-pong in a bidirectional database replication system which includes a plurality of nodes connected via communication media in a topology, each node including a database and a transaction transmitter which sends transactions posted to the database to a database at one or more other nodes for replication in the databases of the one or more other nodes only after the transactions are actually posted to the database of the sending node, each transaction being one or more transaction steps or transaction operations, the computer readable program code means in the article of manufacture comprising:

(a) computer readable program code means for detecting all transactions to be posted to databases in remote nodes that were sent by a local node; and

(b) computer readable program code means for inhibiting the database at the local node from posting selective transactions which were detected as being originally sent by the local node, wherein any transaction sent by a transaction transmitter of a node is replicated in the databases of the one or more other nodes only after the transaction is actually posted to the database of the sending node.

69. An article of manufacture according to claim 68 wherein each of the transactions include transaction data which is usable to identify the origin of the transaction, the bidirectional database replication system further including application code which interprets the transaction data within posted transactions for identifying the origin of the transaction, thereby identifying the originally posted database of the transaction, and the computer readable program code means for inhibiting inhibits the database at the local node from posting selective transactions which were detected as being originally sent by a node having a predetermined origin.

71. An article of manufacture comprising a computer usable medium having computer readable code means therein for selectively preventing transaction ping-pong in a bidirectional database replication system, the system including (i) a first database and (ii) a second database, the system further including (iii) a first transaction transmitter which sends transactions posted to the first database to the second database for replication in the second database, and (iv) a second transaction transmitter which sends transactions posted to the second database to the first database for replication in the first database only after the transactions are actually posted to the second database, each transaction being one or more transaction steps or transaction operations, the computer readable program code means in the article of manufacture comprising:

(a) computer readable program code means for detecting all transactions to be posted to the second database that were sent by the first transaction transmitter; and

(b) computer readable program code means for inhibiting the first database from posting selective transactions which were detected as being originally sent by the first transaction transmitter, wherein any transaction sent by the second transaction transmitter is replicated in the first database only after the transaction is actually posted to the second database.

74. An article of manufacture according to claim 71 wherein the bidirectional database replication system further includes (v) a first transaction receiver which receives transactions sent by the second transaction transmitter for posting to the first database, and (vi) a second transaction receiver which receives transactions sent by the first transaction transmitter for posting to the second database, the computer readable program code means for inhibiting being located in the first transaction receiver.

81. An article of manufacture according to claim 71 wherein the bidirectional database replication system further includes (v) a second transaction receiver which receives transactions sent by the first transaction transmitter for posting to the second database, (vi) a transaction log associated with the second database, and (vii) an audit trail of all transactions posted to the second database, the computer readable code means in the article of manufacture further comprising:

(c) computer readable program code means for assigning a transaction identifier to every transaction received by the second transaction receiver which is posted to the second database;

(d) computer readable program code means for posting the transactions received by second transaction receiver in the second database and creating an audit trail of the posted transactions; and

(e) computer readable program code means for storing the assigned transaction identifiers in the transaction log and associating the assigned transaction identifiers with the transactions in the audit trail, wherein the second transaction transmitter sends the transactions in the audit trail for posting to the first database,

and wherein the computer readable program code means for detecting detects selective transactions in the audit trail which have a transaction identifier similar to a transaction identifier in the transaction log, and the computer readable program code means for inhibiting inhibits the first database from posting to the first database selective transactions in the audit trail which were detected as having a transaction identifier similar to a transaction identifier in the transaction log.

83. An article of manufacture according to claim 82 wherein the bidirectional database replication system further includes (viii) a first transaction receiver which receives transactions sent by the second transaction transmitter for posting to the first database, (ix) a transaction log associated with the first database, and (x) an audit trail of all transactions posted to the first database, the first transaction transmitter sending the transactions in the audit trail of transactions posted to the first database for posting to the second database, the computer readable code means in the article of manufacture further comprising:

(f) computer readable program code means for assigning a transaction identifier to every transaction received by the first transaction receiver which is posted to the first database;

(g) computer readable program code means for posting the transactions received by first transaction receiver in the first database and creating an audit trail of the transactions posted to the first database;

(h) computer readable program code means for storing the assigned transaction identifiers in the transaction log associated with the first database and associating the transaction identifiers assigned by the computer readable program code means for assigning with the transactions in the audit trail of transactions posted to the first database;

(i) computer readable program code means for detecting all transactions to be posted to the first database that were sent by the second transaction transmitter; and

(j) computer readable program code means for inhibiting the second database from posting selective transactions which were detected as being originally sent by the second transaction transmitter,

wherein the computer readable program code means for detecting detects selective transactions in the audit trail of transactions posted to the first database which have a transaction identifier similar to a transaction identifier in the transaction log associated with the first database, and the computer readable program code means for inhibiting inhibits the second database from posting to the second database selective transactions in the audit trail of transactions posted to the first database which were detected as having a transaction identifier similar to a transaction identifier in the transaction log associated with the first database.

84. An article of manufacture according to claim 71 wherein the bidirectional database replication system further includes (v) a second transaction receiver which receives transactions sent by the first transaction transmitter for posting to the second database, (vi) a transaction log associated with the second database, and (vii) an audit trail of all transactions posted to the second database, the second transaction transmitter sending the transactions in the audit trail for posting to the first database, the computer readable code means in the article of manufacture further comprising:

(c) computer readable program code means for preassigning a block of unique transaction identifiers for transactions subsequently received by the second transaction receiver;

(d) computer readable program code means for storing the preassigned block of unique transaction identifiers in the audit trail and in the transaction log;

(e) computer readable program code means for assigning one of the preassigned unique transaction identifiers in the transaction log to each transaction subsequently received by the second transaction receiver; and

(f) computer readable program code means for posting the transactions received by the second transaction receiver in the second database and creating an audit trail of the posted transactions, the audit trail including the assigned transaction identifiers;

wherein the computer readable program code means for detecting detects selective transactions in the audit trail which have a transaction identifier similar to a preassigned transaction identifier stored in the audit trail, and the computer readable program code means for inhibiting inhibits the first database from posting to the first database selective transactions in the audit trail which were detected as having a transaction identifier similar to a preassigned transaction identifier stored in the audit trail.

87. An article of manufacture according to claim 71 wherein the bidirectional database replication system further includes (v) a second transaction receiver which receives transactions sent by the first transaction transmitter, (vi) a second processor which posts the transactions received by the second transaction receiver to the second database, the second processor including a second processor identifier, and (vii) an audit trail of all transactions posted to the second database, the computer readable code means in the article of manufacture further comprising:

(c) computer readable program code means for posting the transactions received by the second transaction receiver in the second database and creating an audit trail of the posted transactions, the audit trail including the second processor identifier, the computer readable program code means for posting being located in the second processor;

wherein the second transaction transmitter sends the transactions in the audit trail to the first database for posting to the first database,

and wherein the computer readable program code means for detecting detects selective transactions which are in the audit trail of the second database and which include the second processor identifier, and the computer readable program code means for inhibiting comprises preventing the first database from posting selective transactions which were detected as being in the audit trail of the second database and which include the second processor identifier.

90. An article of manufacture according to claim 71 wherein the bidirectional database replication system further includes (v) a second transaction receiver which receives transactions sent by the first transaction transmitter, the second transaction receiver including a second transaction receiver identifier, and (vi) an audit trail of all transactions posted to the second database, the computer readable code means in the article of manufacture further comprising:

(c) computer readable program code means for posting the transactions received by the second transaction receiver in the second database and creating an audit trail of the posted transactions, the audit trail including the second transaction receiver identifier,

wherein the second transaction transmitter sends the transactions in the audit trail to the first database for posting to the first database,

and wherein the computer readable program code means for detecting detects selective transactions which are in the audit trail of the second database and which include the second transaction receiver identifier, and the computer readable program code means for inhibiting prevents the first database from posting selective transactions which were detected as being in the audit trail of the second database and which include the second transaction receiver identifier.

93. An article of manufacture according to claim 71 wherein the bidirectional database replication system further includes (v) an audit trail of all transactions posted to the second database, the audit trail including a first audit trail containing all transactions posted to the second database which were sent by the first transaction transmitter, and a second audit trail containing all transactions posted to the second database which were not sent by the first transaction transmitter, the computer readable code means in the article of manufacture further comprising:

(c) computer readable program code means for posting to the second database (i) all transactions sent by the first transaction transmitter and creating an audit trail of the posted transactions in the first audit trail, and (ii) all transactions not sent by the first transaction transmitter and creating an audit trail of the posted transactions in the second audit trail,

wherein the second transaction transmitter sends the transactions in the audit trail to the first database for posting to the first database,

and wherein the computer readable program code means for detecting comprises identifying selective transactions which are in the first audit trail of the second database, and the computer readable program code means for inhibiting prevents the first database from posting selective transactions which were identified as being in the first audit trail of the second database.

94. An article of manufacture according to claim 93 wherein the bidirectional database replication system further includes (vi) a second transaction receiver which receives transactions sent by the first transaction transmitter and posts the transactions which it receives in the second database and creates an audit trail of the posted transactions in the first audit trail,

and wherein the second transaction transmitter sends the transactions in the audit trail to the first database for posting to the first database,

and wherein the computer readable program code means for inhibiting prevents the second transaction transmitter from sending to the first database selective transactions which are in the first audit trail of the second database.

97. An article of manufacture according to claim 71 wherein the bidirectional database replication system further includes (v) a second transaction receiver which receives transactions sent by the first transaction transmitter for posting to the second database, and (vi) an audit trail of all transactions posted to the second database, the computer readable code means in the article of manufacture further comprising:

(c) computer readable program code means for posting the transactions received by the second transaction receiver in the second database and creating an audit trail of the posted transactions; and

(d) computer readable program code means for communicating the transactions received by the second transaction receiver to the second transaction transmitter,

wherein the second transaction transmitter sends the transactions in the audit trail to the first database for posting to the first database,

and wherein the computer readable program code means for detecting comprises detecting selective transactions in the audit trail which were communicated to the second transaction transmitter, and the computer readable program code means for inhibiting comprises preventing the second transaction transmitter from sending to the first database selective transactions in the audit trail which were detected as being communicated to the second transaction transmitter.

98. An article of manufacture according to claim 71 wherein each of the transactions include transaction data which is usable to identify the origin of the transaction, the bidirectional database replication system further including (v) an audit trail of all transactions posted to the second database, and (vi) application code which interprets the transaction data within posted transactions for identifying the origin of the transaction, thereby identifying the originally posted database of the transaction, the computer readable code means in the article of manufacture further comprising:

(c) computer readable program code means for posting to the second database selective transactions sent by the first transaction transmitter and creating an audit trail of the posted transactions, wherein the second transaction transmitter sends the transactions in the audit trail to the first database for posting to the first database; and

(e). computer readable program code means for executing the application code on all transactions read out of the audit trail for sending to the first database,

wherein the computer readable program code means for detecting uses the application code to identify selective transactions which originated at the first database, and the computer readable program code means for inhibiting prevents the first database from posting selective transactions which were identified by the application code as having originated at the first database.

Previous Doc Next Doc Go to Doc#